ELSEVIER

Available online at www.sciencedirect.com

ScienceDirect

Pattern Recognition Letters 28 (2007) 20–30

Pattern Recognition Letters

www.elsevier.com/locate/patrec

Stereovision depth analysis by two-dimensional motion charge memories

José M. López-Valles [a], Miguel A. Fernández [b], Antonio Fernández-Caballero [b,c,*]

[a] *Departamento de Ingeniería Eléctrica, Electrónica, Automática y Comunicaciones, Escuela Universitaria Politécnica de Cuenca, Universidad de Castilla-La Mancha, 16071 Cuenca, Spain*

[b] *Departamento de Sistemas Informáticos, Escuela Politécnica Superior de Albacete, Universidad de Castilla-La Mancha, 02071 Albacete, Spain*

[c] *Instituto de Investigación en Informática de Albacete (I3A), Universidad de Castilla-La Mancha, 02071 Albacete, Spain*

Received 21 October 2005; received in revised form 28 March 2006
Available online 28 July 2006

Communicated by P. Bhattacharya

Abstract

Several strategies to retrieve depth information from a sequence of images have been described so far. In this paper a method that turns around the existing symbiosis between stereovision and motion is introduced; motion minimizes correspondence ambiguities, and stereovision enhances motion information. The central idea behind our approach is to transpose the spatially defined problem of disparity estimation into the spatial–temporal domain. Motion is analyzed in the original sequences by means of the so-called permanency effect and the disparities are calculated from the resulting two-dimensional motion charge maps. This is an important contribution to the traditional stereovision depth analysis, where disparity is got from the image luminescence. In our approach, disparity is studied from a motion-based persistency charge measure.

Keywords: Stereovision; Permanency memories; Disparity analysis; Depth analysis

1. Introduction

In general there are several strategies to retrieve depth information from a sequence of images, such as depth from motion, depth from shading and depth from stereovision. In this paper we introduce a new method to retrieve depth based on motion and stereovision. In a conventional stereoscopic approach, two cameras are usually put together with a horizontal distance between them. As a consequence, objects displaced in depth from the fixation point are projected onto image regions which are shifted with respect to the image center. Due to the geometry of the optic system, it is sufficient to restrict disparity analysis to the projection of corresponding linear segments in the left and the right cameras. In some approaches, the disparity is computed by searching the maximum of the cross-correlation between image windows along the epipolar lines of the left and the right images (Haralick and Shapiro, 1992). This is called the epipolar constraint, which means that for any point in the left image, its matching point in the right image must lie on the corresponding epipolar line.

So far, many algorithms have been developed to analyze the depth in a scene (Koenderink and van Doorn, 1976; Wilson and Knutsson, 1989; Wildes, 1991; Muhlmann et al., 2002; Sumi et al., 2002; Gutiérrez and Marroquın, 2004). Brown et al. (2003) describe a good approximation to all of them in their survey article. According to the correspondence techniques used, we may classify methods into correlation-based, relaxation-based (Grimson, 1985), gradient-based (Choi et al., 2003), and feature-based

[*] Corresponding author. Address: Departamento de Sistemas Informáticos, Escuela Politécnica Superior de Albacete, Universidad de Castilla-La Mancha, 02071 Albacete, Spain. Tel.: +34 967 599200; fax: +34 967 599224.

E-mail address: caballer@info-ab.uclm.es (A. Fernández-Caballero).

0167-8655/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2006.05.009

(Venkateswar and Chellappa, 1995). The main correlation-based technique is the area-correlation technique (e.g., (Zabih and Woodfill, 1994)). Area-based approaches have the advantage of providing a more robust correspondence analysis, in opposite to pixel-based approaches that generate directly dense disparity maps. Matching elements for area-based methods are the individual pixels over which the matching cost is evaluated; pixel-to-pixel correspondence is evaluated on image intensity function and similarity statistics. For example, a work (Binaghi et al., 2004) investigates the potential of neural adaptive learning to solve the correspondence problem within a two-frame adaptive area matching approach. The method is based on the use of the zero mean normalized cross-correlation coefficient integrated within a neural network model which uses a least-mean-square delta rule for training. Another approach (Di Stefano et al., 2004) proposes an area-based stereo algorithm suitable for real time applications, where the core of the algorithm relies on the uniqueness constraint and on a matching process that rejects previous matches as soon as more reliable ones are found.

After about 40 years of research on computational stereo there are still open problems, such as global correspondence and methods for handling occlusion. The most significant advance has been the appearance of real-time stereo systems; however real-time algorithms, are still relatively simplistic, and most of the global matching and occlusion handling methods do not currently run in real-time (Brown et al., 2003). In this paper an area-correlation-based method that turns around the existing symbiosis between stereovision and motion is introduced; motion minimizes correspondence ambiguities, and stereovision enhances motion information. This symbiosis has been painstakingly studied to get a major performance in our artificial three-dimensional disparity depth analysis of moving non-rigid objects through stereovision. Our conviction is that working only on moving objects is of a great importance to gain reliability in correspondence analysis, occlusion handling and real-time implementation. Most methods have in common that they work with static images and not with motion information, although some other approaches have already been introduced (Ho and Pong, 1996; Liu and Skerjane, 1993; Xu, 1995). More recently, in (Zhang et al., 2003) the traditional binocular stereo problem is extended into the space–time domain, in which a pair of video streams is matched simultaneously instead of matching pairs of images frame by frame. By utilizing both spatial and temporal appearance variation, this modification reduces ambiguity and increases accuracy. Also recently, it has been shown that methods derived from the space–time stereo framework can be used to recover depth in situations in which existing methods perform poorly (Davis et al., 2005). Also, a separate edge-preserving regularization scheme to calculate disparity fields for a stereoscopic image pair and a joint disparity and motion estimation algorithm for stereoscopic video sequences has been presented (Yang et al., 2005).

The motion analysis algorithm used in this work has already been tested in applications such as moving object shape recognition in noisy environments (Fernández-Caballero et al., 2003a,b,c), moving objects classification by motion features such as velocity or acceleration (Fernández et al., 2003), and in applications related to selective visual attention (Fernández-Caballero et al., 2004; López et al., 2006). Our proposal is to analyze motion in the original sequences by means of the so-called permanency effect (Fernández et al., 2003) and to analyze the disparities from the resulting charge maps. As a novelty, in this paper motion analysis performs separately on both stereovision sequences. Thus, the central idea behind our approach is to transpose the spatially defined problem of disparity estimation into the temporal domain and compute the disparity simultaneously with the incoming data. This is an important contribution to the traditional disparity analysis, where disparity is got from the image luminescence. In the present approach, disparity is studied from a timely consistent persistency charge measure. The objective of our method is to calculate the depth of the moving elements present in a video sequence by studying the correspondence of right and left image objects with a similar motion history, and eliminating any superfluous information of the static elements. Some very important application areas for our stereovision depth analysis method are visual surveillance and autonomous (vehicle and robot) navigation.

The rest of the paper is structured as follows. Our *Stereovision Depth Analysis* method is described in Section 2. In Section 3, experimental results on a real video sequence are described. Finally conclusions are given in Section 4.

2. Description of the stereovision depth analysis

The computational structure which supports the *Stereovision Depth Analysis* can be seen in Fig. 1 and is described next. This structure is the result of the analysis of the stereovision's geometric problem and the application of the pertinent restrictions, as well as the study of biological stereovision systems and the permanence mechanisms, which our research group is very familiar with. Next, the proposed structure is described in a general way, dedicating the rest of the section to the detailed description of each subtask in which the solution to the problem is broken down.

The input to our system is a pair of stereo image sequences. These sequences have been acquired by means of two cameras arranged in a parallel configuration. In a well-calibrated fronto-parallel camera arrangement the epipolar lines are horizontal and thereby identical to the camera scan-lines. Thus, they will capture two similar, although not exactly equal, scenes. In case the images have been acquired in a convergent configuration, horizontal epipolar lines can be obtained by image rectification techniques (Faugeras, 1993).

Firstly (see Fig. 1), frame by frame, the permanence or accumulative computation effect (Fernández et al., 2003; Fernández-Caballero et al., 2003a,b,c; López et al., 2006)

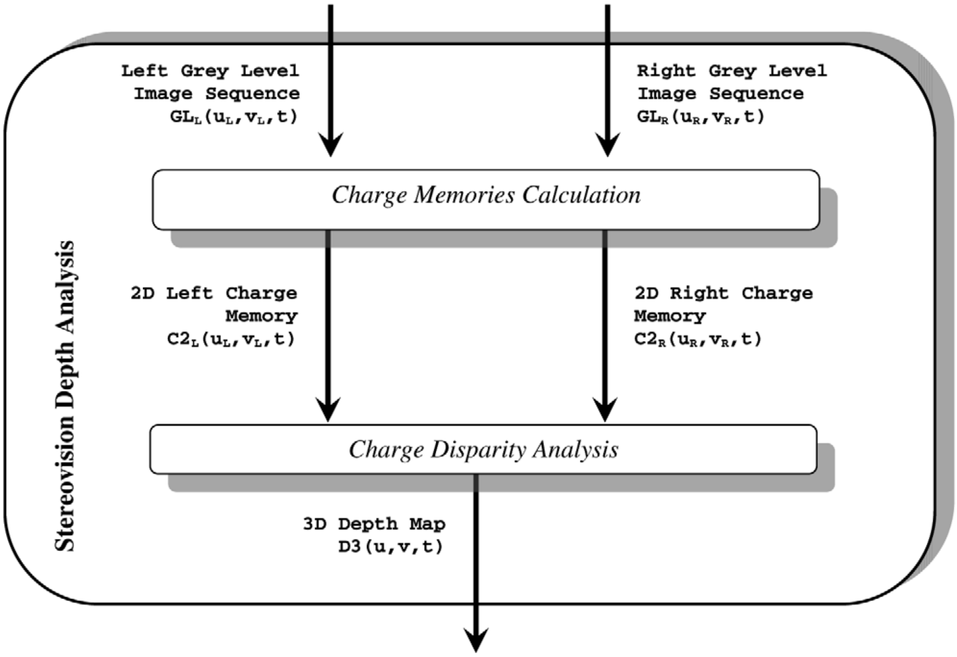
Left Grey Level Image Sequence
$GL_L(u_L,v_L,t)$
Right Grey Level Image Sequence
$GL_R(u_R,v_R,t)$
Charge Memories Calculation
2D Left Charge Memory
$C2_L(u_L,v_L,t)$
2D Right Charge Memory
$C2_R(u_R,v_R,t)$
Charge Disparity Analysis
3D Depth Map
$D3(u,v,t)$
Stereovision Depth Analysis

Fig. 1. Complete outline of "Stereovision Disparity Analysis".

is applied for the purpose of *Charge Memories Calculation* to the stereo image pairs. Let (u_L, v_L) and (u_R, v_R) be the spatial coordinates of the left and right images, respectively, and let t be the temporal coordinate in our system. The input is the stereo image pair in grey levels, $GL_L(u_L, v_L, t)$ and $GL_R(u_R, v_R, t)$, corresponding to a frame and the output to each frame is the state of a couple of two-dimensional charge memories; that is, a *2D Charge Memory* for the left image, $C2_L(u_L, v_L, t)$, and another one for the right image, $C2_R(u_R, v_R, t)$. Then *Charge Disparity Analysis* carries out the matching process between both right and left sequence charge memories for each frame. The output is a depth memory per frame, $D3(u, v, t)$. Notice that (u, v) are the spatial coordinates of D3. In general, (u, v) corresponds to (u_L, v_L) in our way of processing the scene depth, although (u, v) could be calculated from reference (u_L, v_L). Well-known concepts in stereovision will be applied in this step, such as restrictions to the correspondences and primitives, except that in our case applied to the charge memories obtained in the previous section instead of object–shape information as usual.

2.1. Charge memories calculation

The purpose of *Charge Memories Calculation* is to represent two-dimensional motion for every input sequence in the permanence elements' charge levels. Fig. 2 shows the whole process as well as the interrelations between the sub-processes.

Two parallel processes are observed; each one belongs to an input sequence, right and left. From there, the *Grey Level Bands Segmentation* subsystem separates each of the frames into related regions for the purpose of later analyzing its movements. The motion detection system, by means of permanency, requires as its input, the current image segmented into grey level bands, as well as the previous image. The purpose of this is to analyze which memory elements have skipped between the bands, detecting movement in the corresponding pixels.

2.1.1. Grey-level bands segmentation

The *Grey-Level Bands Segmentation* subtask transforms the input images captured in 256 grey levels into a lower number of levels (nGLB). In general, the use of 8 levels produces good results as demonstrated in previous works by some of the same authors for monocular sequences (e.g. Fernández-Caballero et al., 2003a,b,c). These nGLB-level images are called grey level band segmented images $(GLB_{L/R})$,[1] and they are calculated as

$$GLB_{L/R}(u_{L/R}, v_{L/R}, t) = \left\lfloor \frac{GL_{L/R}(u_{L/R}, v_{L/R}, t) \cdot n\mathrm{GLB}}{GL_{\max} - GL_{\min} + 1} \right\rfloor + 0.5 \tag{1}$$

where $GL_{\max}$ and $GL_{\min}$ are the maximum and minimum grey-level values, respectively, for the input image. Typically, $GL_{\max} = 255$ and $GL_{\min} = 0$ in all our implementations. In Eq. (1), the expression $GL_{\max} - GL_{\min} + 1$ corresponds to the number of input levels, which will generally be 256.

Obviously, segmentation in grey level bands is just one possible clustering or image segmentation algorithm. The

[1] Notice that throughout the whole text "L/R" is equivalent to "L or R".

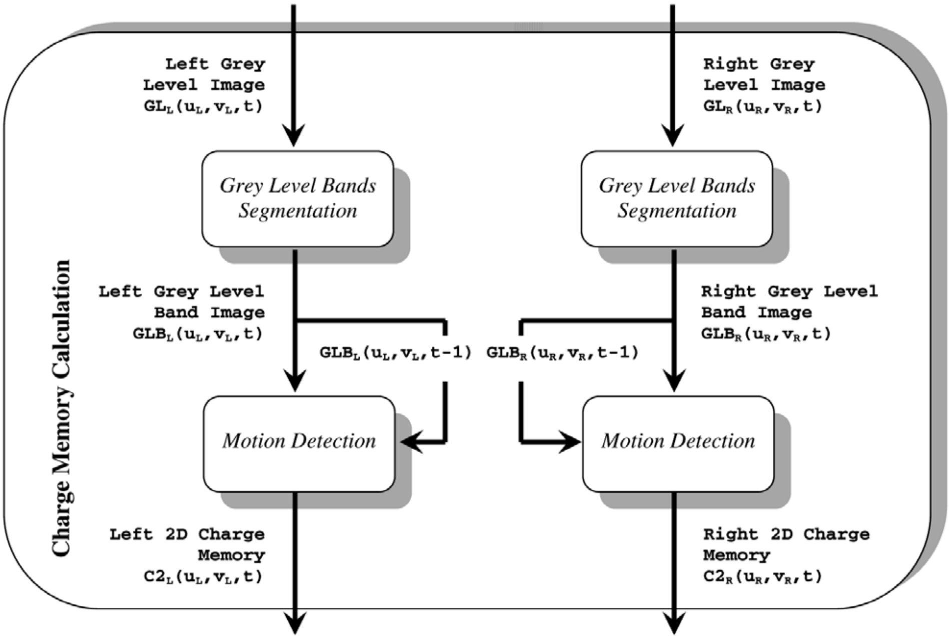

Fig. 2. "Charge Memory Calculation" process outline.

two reasons why we prefer to work with grey-level bands are: (a) A traditional movement detection system is based on image difference. By matching one grey-level category in one single band, into considering that there is movement when a variation in the grey-level band is detected, the noise level is reduced due to small brightness variations in a single object between two consecutive images. (b) There is also a decrease in the computational complexity, noting the great parallelism used in the algorithms of the proposed model. We go on to parallel computing in the order of number of grey-level bands nGLB, and not grey levels nGL, with $n\text{GL} > n\text{GLB}$. In comparison to a well-known technique such as k-means, our proposal may be considered as computationally inexpensive.

Up to this moment, we have only performed a simple scale change. A detailed analysis of the features and performances of this segmentation method is not the purpose of this paper; a good description may be found in (Fernández-Caballero et al., 2003a,b,c). Notice that we are not yet deciding whether there is variation or not in the (u, v) point's grey-level band. To sum up, the result of the *Grey-Level Bands Segmentation* subtask is, for each input image pixel, the grey-level transformation in its corresponding grey-level band. That is, the result will be a matrix of the same size as the input image, but its content will have values between 1 and nGLB.

2.1.2. Motion detection

Once the grey-level bands for each pixel in both images have been established, the next step is motion characterization. For this, we first establish a two-dimensional motion memory for each sequence, which will be updated for each frame. This two-dimensional motion memory will have as many motion detection elements as pixels are in the horizontal and vertical dimensions of the images.

The *Movement Presence*, $M2_{\text{L/R}}(u_{\text{L/R}}, v_{\text{L/R}}, t)$, is obtained through pixel by pixel comparison of two consecutive images segmented into grey-level bands. If pixel $(u_{\text{L/R}}, v_{\text{L/R}})$ at a moment of time t belongs to the same grey-level band than at moment of time $t-1$, then there has been no movement; whereas, if there has been a change in $\text{GLB}_{\text{L/R}}$, then it is assumed that there has been movement.

$$M2_{\text{L/R}}(u_{\text{L/R}}, v_{\text{L/R}}, t) = \begin{cases} 0, & \text{if } \text{GLB}_{\text{L/R}}(u_{\text{L/R}}, v_{\text{L/R}}, t) = \text{GLB}_{\text{L/R}}(u_{\text{L/R}}, v_{\text{L/R}}, t-1) \\ 1, & \text{if } \text{GLB}_{\text{L/R}}(u_{\text{L/R}}, v_{\text{L/R}}, t) \neq \text{GLB}_{\text{L/R}}(u_{\text{L/R}}, v_{\text{L/R}}, t-1) \end{cases} \quad (2)$$

These two-dimensional motion presence charge memories identify those input image pixels where a jump between grey-level bands has occurred and thus, the image points for coordinates $(u_{\text{L/R}}, v_{\text{L/R}})$ where there has been movement.

With the purpose of obtaining more accurate information about movement, it is convenient to detect not only the points where there has been movement, but also the movement's more recent history. For this reason, we use the permanence charge memories. By means of accumulative computation mechanisms on the negation of the *Motion Presence* characteristic, this subtask obtains the *2D Motion Charge Memory* associated with the accumulation process, as shown in Eq. (3):

$$C2_{\text{L/R}}(u_{\text{L/R}}, v_{\text{L/R}}, t) = \begin{cases} C2_{\max}, \\ \quad \text{if } M2_{\text{L/R}}(u_{\text{L/R}}, v_{\text{L/R}}, t) = 1 \\ \max[C2_{\min}, C2_{\text{L/R}}(u_{\text{L/R}}, v_{\text{L/R}}, t-1) - C2_{\text{dis}}], \\ \quad \text{if } M2_{\text{L/R}}(u_{\text{L/R}}, v_{\text{L/R}}, t) = 0 \end{cases} \quad (3)$$

$C2_{\min}$ and $C2_{\max}$ are the minimum and maximum values, respectively, that the values stored in the two-dimensional charge memory (*2D Motion Charge Memory*, $C2_{L/R}(u_{L/R}, v_{L/R}, t)$) can reach. $C2_{\text{dis}}$ is the memory discharge value. When giving values to variable $C2_{\text{dis}}$, we must consider the sampling-speed ratio of the sequences captured. The idea behind this process is that if no movement exists in point $(u_{L/R}, v_{L/R})$, that is to say, when $M2_{L/R}(u_{L/R}, v_{L/R}, t) = 0$, then the charge value $C2_{L/R}(u_{L/R}, v_{L/R}, t)$ will decrease until it reaches $C2_{\min}$. If movement does exist ($M2_{L/R}(u_{L/R}, v_{L/R}, t) = 1$), then the complete charge takes place, taking the $C2_{\max}$ value. Thus, a point in which there has been recent movement will have charge values between saturation ($C2_{\max}$) and complete discharge ($C2_{\min}$), but different from them. The values will be closer to saturation, the more recent the movement is; inversely, the values will be lower, the longer it has been since movement took place in this area of the image. This way, the charge value is proportional to the time that has elapsed since the last significant brightness variation for each pixel of the image.

Fig. 3 shows all these issues. Fig. 3a and b shows two images of a monocular sequence. The advance of a car may be noticed, as well as a more slight movement of a pedestrian. In Fig. 3c you may observe the effect of these moving objects on the permanence memory drawn as a two-dimensional image.

The difference between a quick object as it is the car, which is leaving a very long motion trail (from dark grey to white), and a pedestrian whose velocity is clearly slower and whose motion trail is nearly unnoticeable with respect to the car's one, is presented. Thus, permanency memories enable to represent the motion history of the frames that form the image sequence, and it is possible to segment from the motion of the objects present in the scene. For a more extended description of the permanence effect, see (López et al., 2006).

2.2. Charge disparity analysis

The output of *Charge Disparity Analysis* is a three-dimensional depth memory, $D3(u, v, t)$, which shows the depth of the points in the scene where there has been movement. Using the charge memories as input has two important advantages. In the first place, only information about motion is used, filtering out all static information from the scene, whether it is 2D or 3D. Since our objective is to obtain a three-dimensional memory of the scene's motion, it is more an advantage than a disadvantage to have a filtered memory, as static elements only contribute to noise for this project. On the other hand, object motion leaves similar charge trails in both permanence charge memories. Thus, matching of the moving objects trails in the sequence will be simpler and more robust.

In order to explain our disparity analysis method, it is sufficient to analyze the process at the level of epipolar lines. The key idea is that a moving object causes two identical trails to appear in epipolar lines of the permanency stereo-memories. The only difference relies on their relative horizontal positions, affected by the disparity of the object at each moment. In Fig. 4, a simple example is offered, where the charge values in two corresponding superimposed epipolar lines of the memories are represented. In a parallel configuration as the one we have chosen, there will be no disparity in the right and the left images for objects that are in a great depth – imagine in the infinite. Nevertheless, when an object approaches to the central point of the base line, that is to say, between the two cameras, the object goes appearing more to the right on the left image and more to the left on the right image. This is precisely the disparity concept; the more close objects have a greater disparity than the more distant ones.

So, looking at Fig. 4 it is possible to analyze the motion of each one of the three objects present in the permanency memories (or charge memories) from their motion trails. You may observe that object "a", which has a long trail and has its maximum charge towards the left, is advancing

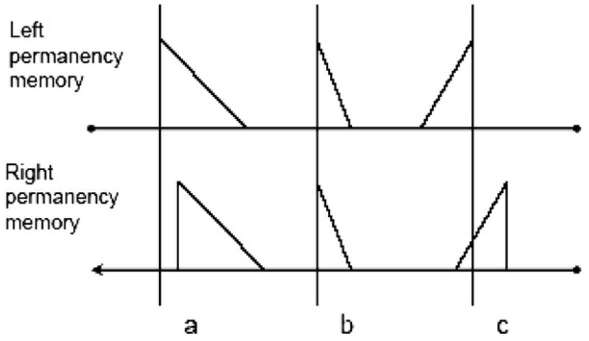

Fig. 4. Disparity by motion charge (permanency) memories.

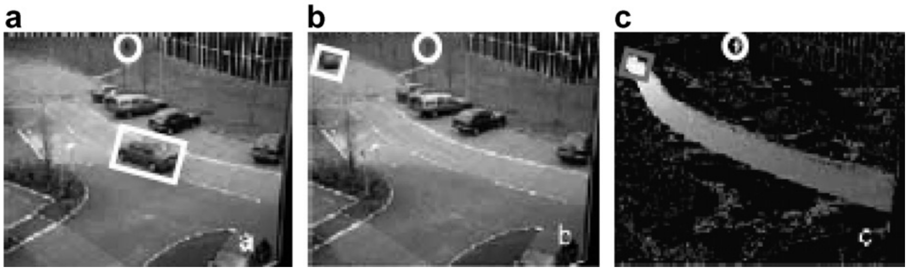

Fig. 3. Motion charge Memory: (a) one image of a sequence, (b) same perspective after some seconds, (c) motion trails as represented on the two-dimensional motion charge map.

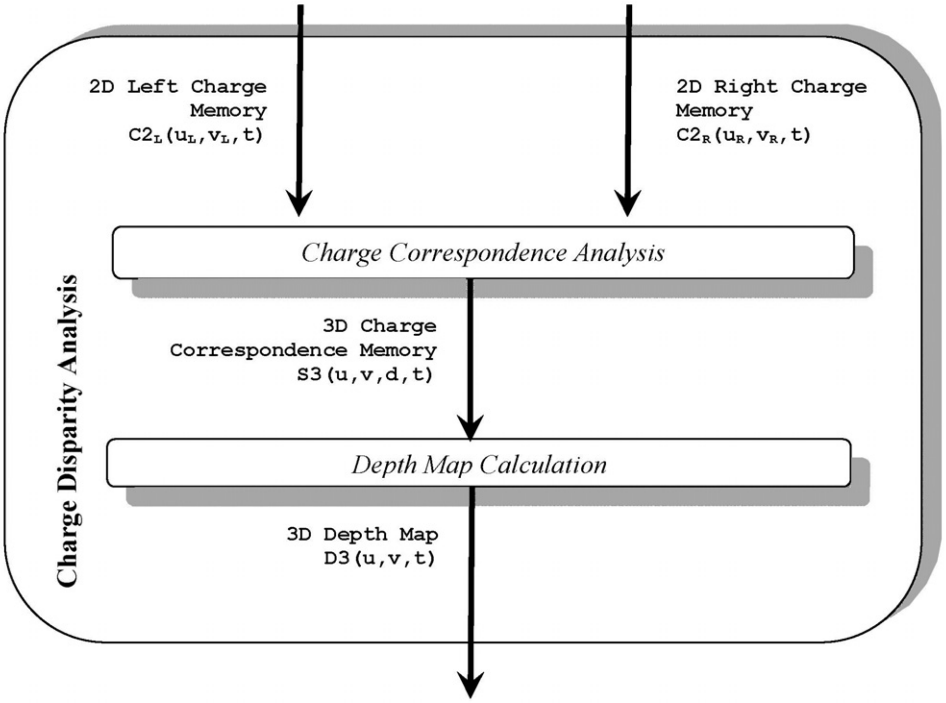

Fig. 5. "Charge Disparity Analysis" process outline.

to the left at a high speed. Object "b", with a shorter trail, is also advancing towards the same direction but at a slower velocity. Finally, object "c", whose trail is inverted in horizontal, is moving to the right at a medium velocity, as shown by its trail.

This simple example draws two main conclusions. Firstly, in order to consider two motion trails to be correspondent, it must only be checked that both are equal enough in length and in discharge direction in epipolar lines of the permanency stereo-memories. Secondly, we may state that, in order to analyze disparities, one possibility is to displace one epipolar line over the other one, until we get the exact point where both lines are completely superimposed. In other words, an epipolar line has to be displaced over the other until motion trails maximally coincide. Of course, the right epipolar line can be displaced over the left or the left epipolar line over the right. When the motion trails coincide (with a given error ratio or threshold), the displacement value applied to the epipolar line is precisely the searched disparity value.

But, objects are not usually uniform and the layout of the permanence memories is much more complex. This is why an object seen as the set of its component parts shows various disparities. Thus, it is necessary to analyze the correspondence from the values of the various parts of the objects to obtain one valid overall disparity value. The most efficient way to manage this is that each pixel obtains its disparity in such a way that the maximum of its neighboring charge values confirm a consensus disparity.

All these considerations tell us that the disparity analysis at epipolar line level consists in superimposing both epipolar lines with different relative displacements and in analyzing the correspondences produced in the neighborhood of each charge unit. The displacement which produces that a maximum number of surrounding elements confirm its correspondence is assigned the more trustful disparity value. Precisely, in Fig. 5, the process diagram for this subtask is shown. In it, we can also see the interrelation between the processes *Charge Correspondence Analysis* and *Depth Memory Calculation*, as well as the inputs and outputs involved in each process. The correspondence analysis output will be called *3D Charge Correspondence Memory*, $S3(u,v,d,t)$, where d stands for disparity. From this charge correspondence memory, the maximum reliability depth for each coordinate (u,v) will be decided. This will be done by means of the *3D Depth Memory Calculation* process. The output will be called *3D Depth Memory*, $D3(u,v,t)$.

According to the taxonomy proposed by Scharstein and Szeliski (2002), a dense stereo matching process, as the one performed in our proposal, can be divided into three tasks: *matching cost computation*, *aggregation of local evidence* and *computation of disparity values*. Local methods usually compute final disparity adopting a local winner-take-all strategy which selects the pair with the best matching cost under assumption of uniqueness. This is also the overall approach taken in our work. *Charge Correspondence Analysis* performs the matching cost computation and the aggregation of local evidence, whereas *Depth Memory Calculation* computes the disparity values.

2.2.1. Charge correspondence analysis

The purpose of this subtask is to prepare the necessary information to decide on the disparity with the greatest reliability for each of the processing elements in the input charge memory for *3D Depth Memory Calculation*. This task will mainly take the epipolar, the ordering and the disparity restrictions into account.

It is common knowledge that the most robust correspondence primitives are those with the highest contrast, such as contours or regions. In our case, we intend to carry out a correspondence analysis per region (area-based disparity calculation), and we name them *constant disparity* regions. Therefore, we must group together those neighboring charge elements whose corresponding elements have the same (or very similar) disparity. But before grouping the neighboring elements together, remember what we call corresponding charge elements. It is, basically, a question of finding which pixel has a similar history of movement in the opposite epipolar line and, consequently which processing element for the corresponding charge memory has an instant charge level stored with a similar value.

In the next subsections we introduce our proper proposal for analyzing the charge correspondences, as an alternative to some existing methods as dynamic programming, intrinsic curves, graph cuts, etc. The reliability criterion to be chosen will depend on the position of each processing element and it will have to do with the size of each *constant disparity* region. This size is calculated in two phases: (a) First of all, we carry out a horizontal counting of all adjoining neighbors that belong to this region. (b) Afterwards, the vertical values found for all the adjoining vertical processing elements, which also belong to this region, are accumulated. In the following subsections, all the related phases, that is to say, *Pixel-wise Charge Correspondence Analysis*, *Horizontal Charge Counting and Homogenizing*, and *Vertical Charge Accumulation and Homogenizing*, are going to be explained.

2.2.1.1. Pixel-wise correspondence analysis. The *Pixel-wise Charge Correspondence Analysis* on the charge elements of both corresponding *2D Motion Charge Memories*, $C2_{\mathrm{L/R}}(u_{\mathrm{L/R}}, v_{\mathrm{L/R}}, t)$ is carried out. By applying the epipolar restriction, each charge element from a charge memory is compared to those from the other charge memory on the same row, although displaced horizontally up to the maximum limit $d_{\max}$, imposed by the disparity restriction. The calculation expression for the elements of this three-dimensional output matrix is as follows:

$$Sa3(u,v,d,t) = \begin{cases} 1, & \text{if } |C2_L(u,v,t) - C2_R(u+d,v,t)| \leqslant \theta a3 \\ 0, & \text{otherwise} \end{cases}, \quad \forall d | 0 \leqslant d \leqslant d_{\max} \tag{4}$$

The three-dimensional matrix $Sa3(u,v,d,t)$ indicates whether or not there is a specific correspondence (similarity in charge above threshold value $\theta a3$) between both charge memories for each coordinate (u,v) and for each disparity value d.

2.2.1.2. Horizontal charge counting and homogenizing. The purpose in this second step is to establish a charge elements' matrix the same size as the three-dimensional matrix $Sa3$ from the previous step, where each element would have the amount of horizontally corresponding adjacent input elements as the final result (those where $Sa3(u,v,d,t) = 1$). This process is carried out in two phases: the first (*Horizontal Charge Counting*) running to the left counts the input elements set on 1 and stores the value in the corresponding output charge element ($Sb3(u,v,d,t)$).

for $u = 1$ to image_width

$$Sb3(u,v,d,t) = \begin{cases} Sb3(u-1,v,d,t)+1, & \text{if } Sa3(u,v,d,t) = 1 \\ 0, & \text{if } Sa3(u,v,d,t) = 0 \end{cases} \tag{5}$$

Once the horizontal counting to the right is done, a charge homogenizing is carried out (*Horizontal Charge Homogenizing*) so that all the charge elements belonging to a horizontal constant disparity region acquire the same charge value. This value corresponds to the horizontal size of the constant disparity region, formed by all of them.

for $u = \text{image\_width} - 1$ downto 1

$$Sc3(u,v,d,t) = \begin{cases} \max[Sc3(u+1,v,d,t), Sc3(u,v,d,t)], \\ \quad \text{if } Sb3(u,v,d,t) \geqslant 1 \\ 0, \\ \quad \text{if } Sb3(u,v,d,t) = 0 \end{cases} \tag{6}$$

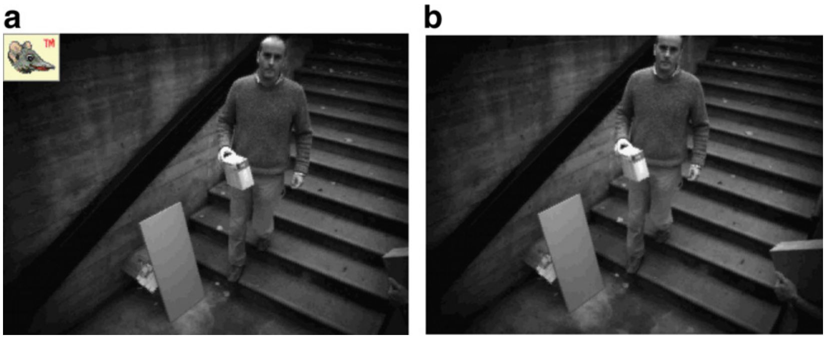

Fig. 6. Frame # 211 from sequence "OutdoorZoom". (a) Left input image. (b) Right input image.

2.2.1.3. Vertical charge accumulation and homogenizing. On this third step, we aim to establish a new charge element matrix the same size as the three-dimensional matrix $Sc3$ from the previous step, where each element has, as final result, the charge accumulation from the neighboring input elements, which are considered vertically correspondent. This process is done, as in the previous case, in two periods: a first running downwards (rising v values) counts the input elements other than 0 and stores the accumulated value in the corresponding output charge element ($Sd3(u,v,d,t)$).

for $v = 2$ to image_height

$$Sd3(u,v,d,t) = \begin{cases} Sd3(u,v,d,t) + Sd3(u,v-1,d,t), \\ \quad \text{if } Sc3(u,v,d,t) \geqslant 1 \\ 0, \\ \quad \text{if } Sc3(u,v,d,t) = 0 \end{cases} \tag{7}$$

Once the counting towards positive v values is done, charge homogenizing is carried out in such a way that all charge elements belonging to a constant disparity region have vertically the same charge value. This value will correspond to a region's total size.

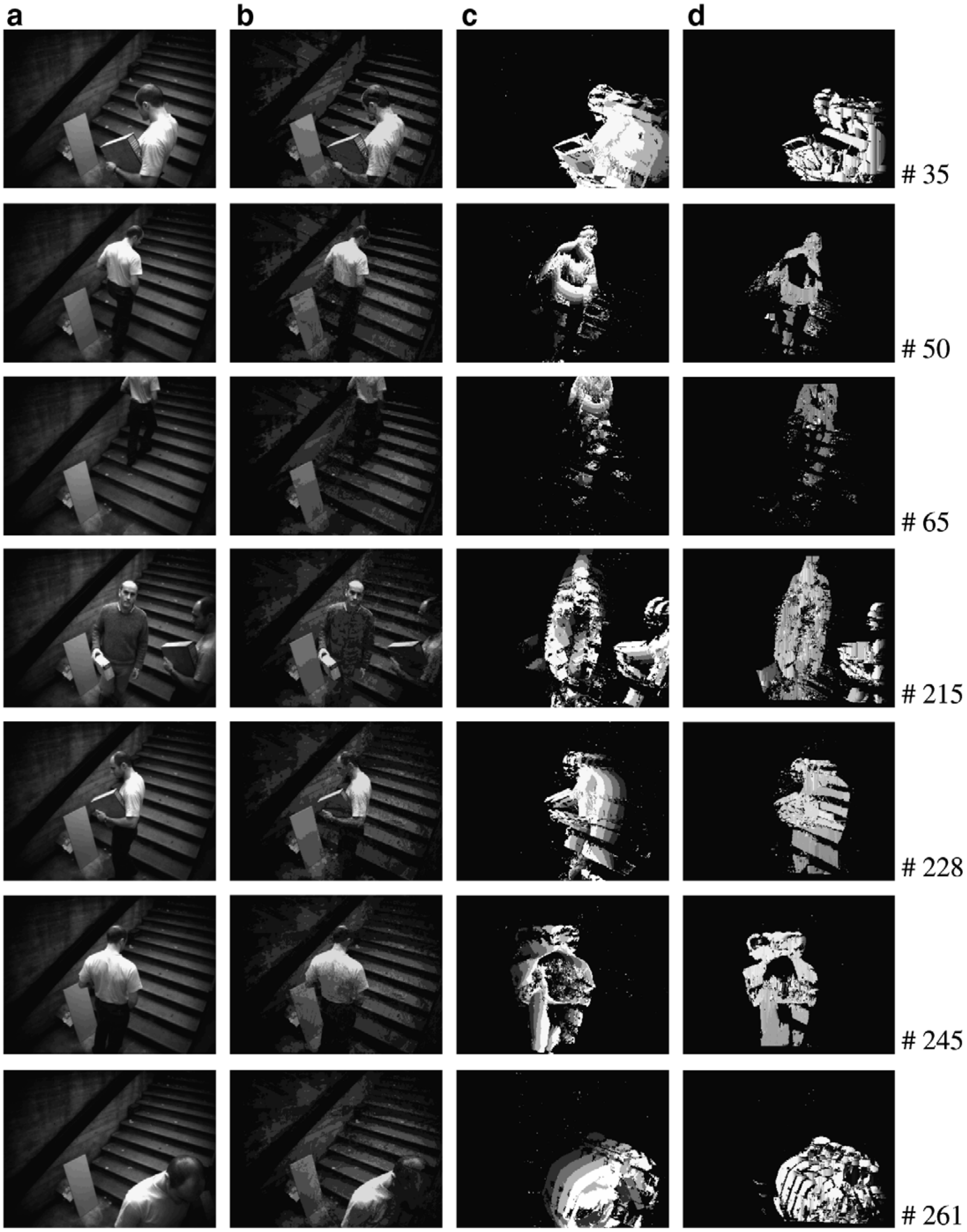

Fig. 7. Results for the “OutdoorZoom” scenario for several frames. (a) Right input image. (b) Right image segmented into grey-level bands. (c) Right 2D Charge Memory. (d) 3D Depth Map.

for $v = \text{image\_height} - 1$ downto 1

$$S3(u,v,d,t) = \begin{cases} \max[S3(u,v,d,t), S3(u,v+1,d,t)], \\ \quad \text{if } Sd3(u,v,d,t) \geqslant 1 \\ 0, \\ \quad \text{if } Sd3(u,v,d,t) = 0 \end{cases} \tag{8}$$

2.2.2. 3D depth memory calculation

Once we have calculated the region's sizes which each charge memory correspondence belongs to, we need to associate, as maximum reliability disparity for each pixel, those values whose charge $S3(u,v,d,t)$ is maximal in d. With this, we are imposing the uniqueness restriction, since each processing element will only have a single disparity value as a final value.

The processing carried out to obtain the disparity associated to each charge element is shown in the following expression:

$$D3(u,v,t) = i | S3(u,v,i,t) \geqslant S3(u,v,j,t), \\ \forall(i,j), 0 \leqslant i,j \leqslant d_{\max} \tag{9}$$

This operation calculates the value i whose $S3(u,v,i,t)$ is maximum in the third dimension. The ordering restriction is included in the method proposed for the charge disparity analysis subtask, since the specific correspondence verification and subsequent region configuration means maintaining the order of the correspondences found. Based on the charge disparity calculation done, we can estimate the moving elements' depth. With this, we have obtained a stereoscopic motion memory in which each moving element from the scene appears associated to its depth.

3. Data and results

In order to test our algorithms, the results of their application are shown on a real stereo sequence, called "OutdoorZoom" (see Fig. 6), downloaded from http://labvisione.deis.unibo.it/~smattoccia/stereo.htm.

The whole sequence is 30 s long and has been acquired at a rate of 10 images per second. This is precisely a stereo sequence created and used by the authors of (Di Stefano et al., 2004) within a research activity aimed at developing a 3D People Tracking application. The tracking approach is based on first merging disparity maps with the information provided by a grayscale change-detection algorithm and then building a suitable plane-view representation that enables to track moving objects in the 3D space. Notice that our results have been obtained without any information related to the calibration of the cameras. Our algorithms may use any stereo video sequences, as we do not try to solve the geometric analysis of more traditional correspondence analysis algorithms (Faugeras, 1993). The values of the main parameters used in our test series were

$$C2_{\text{dis}} = 128;\ n\text{GLB} = 8;\ C2_{\min} = 0; C2_{\max} = 255; \theta a3 = 1$$

Fig. 7 shows the result for some of the more representative results of applying our algorithms to the "OutdoorZoom" scenario. In column (a) the input image is shown, in column (b) the segmentation in grey-level bands may be noted, in column (c) motion information as represented in the right motion charge map is offered, and in column (d) the final output, that is to say, the scene depth as detected by the algorithms, is presented. You may observe that the light colors in column (d) means that people are closer to the cameras. Black means there is no motion detected. The main information is available in columns (c) and (d). We may see some details, as, for example, the following ones:

- In frame 35, a person is entering the scene on the right side, very close to the cameras. This is why in column (d), the final output, very light grey levels appear.
- This person is progressively moving away from the cameras, in such a way that on frame 50 it is represented by intermediate grey levels.
- In frame 65, the person is now far away from the cameras. Its shape appears in dark-grey values.

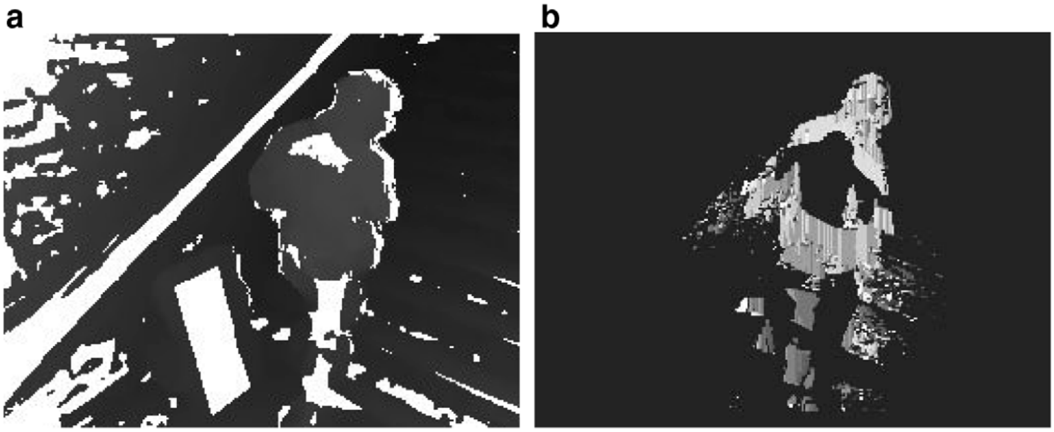

Fig. 8. Results for the "OutdoorZoom" on frame 50. (a) Di Stefano disparity map with threshold set to 3. (b) Our depth map.

- Let us now focus on frame 215. A person is walking down the steps and at the same time an object is appearing on the right side of the image. It may be noticed at the output of the system that the object is a bit lighter than the person. Thus, the object has to be closer to the cameras than the walking person.
- From frame 215 to frame 228, the pedestrian is walking horizontally (to the left). Thus, we see no difference in the grey levels present in these frames.
- In frame 245, the person turns around, but there is still no noticeable difference in its depth in the scene.
- Lastly, in frame 261, we may observe the person leaving the scene on the right side, and very light grey levels at the output. This obviously means that the man is very close to the cameras.

The only possible comparison for the "OutdoorZoom" scenario is with the results of (Di Stefano et al., 2004). As the aim of both approaches is totally different – Di Stefano and collaborators work on luminescence parameters on the whole image, whereas we compute on the persistency of moving objects – it is only possible to provide a qualitative impression of the reliability of our algorithms (see Fig. 8).

As opposed to applying a matching algorithm on static images, our approach is to work with the motion history present in the whole video sequence. The aim of Di Stefano and colleagues is to perform object tracking, whilst our approach is to analyze the depth of moving objects. When looking at Fig. 8a, you may appreciate that Di Stefano et al. segment all objects of the scene quite well. In our case (Fig. 8b), only the person moving in the scene is segmented, and the depth is obtained. According to this reasoning, we may state that our stereovision depth analysis is very accurate for the objective of calculating the depth of the moving elements present in a video sequence from the correspondence of right and left image objects with a similar motion history.

4. Conclusions

According to the conclusions provided by (Gutiérrez and Marroquín, 2004), the matching approach to solve the so-called correspondence problem in stereovision has intrinsic limitations. Hence, the stereovision depth analysis problem is still a bug challenge in computer vision. In this paper a new method to retrieve disparity information based on motion and stereovision has been introduced. A motion detection representation helps to establish further correspondences between different motion information. This representation is based on the permanency memories mechanism, where jumps of pixels between grey level bands are computed in a matrix of charge accumulators. Thus, for the purpose of analyzing scene depth from stereo images, we have chosen the alternative of not using direct information from the image, but rather the one derived from motion analysis. This option provides an important advantage. It is easier to use correspondences through motion information stored in two-dimensional motion charge maps than by grey level information of the frames. When observing motion features of a particular object in both stereo sequences at the same time instant, we notice that these features are extremely similar. This is the reason why it is easy and robust to establish correspondences between the motion information of an object at the right image respect to the object at the left image. The number of ambiguity possibilities is lower, as we have filtered a lot of information by eliminating all static elements in the scene. By reducing the number of elements (what does not move, does not exist), the matching process is also easier. Moreover, information about motion is associated to the history of the last few frames in the scene, not only to the present scene. Thus, the matching process takes place based on more accurate information (two similar elements with different recent motion history generate different trails).

Charge memory displacement allows us to transform disparity into time, time into charge, and finally, charge into depth. The proposed solution involves a type of process which tries to take advantage only of the use of high-order primitives and pixels. On the one hand, the elements placed in correspondences are regions obtained from moving objects' motion trails by means of permanence memory interpretation. On the other hand, the fact that each pixel can decide, through a local analysis and based on motion trail overlapping, with a more reliable disparity, creates a dense disparity memory, which is considered a great advantage in pixel-based correspondence systems.

Acknowledgements

This work is supported in part by the Spanish CICYT TIN2004-07661-C02-02 grant. The authors are grateful for the "OutdoorZoom" sequence available at http://labvisione.deis.unibo.it/~smattoccia/stereo.htm. We also appreciate very much the very interesting suggestions provided by the anonymous referees, which have greatly enhanced this paper.

References

Binaghi, E., Gallo, I., Marino, G., Raspanti, M., 2004. Neural adaptive stereo matching. Pattern Recognition Lett. 25, 1743–1758.

Brown, M.Z., Burschka, D., Hager, G.D., 2003. Advances in computational stereo. IEEE Trans. Pattern Anal. Machine Intell. 25 (8), 993–1008.

Choi, I., Yoon, J.-G., Lee, Y.-B., Chien, S.I., 2003. Stereo system for tracking moving object using log-polar transformation and zero disparity filtering. In: Proc. 10th Internat. Conf. on Computer Analysis of Images and Patterns, pp. 182–189.

Davis, J., Nehab, D., Ramamoorthi, R., Rusinkiewicz, S., 2005. Space–time stereo: A unifying framework for depth from triangulation. IEEE Trans. Pattern Anal. Machine Intell. 27 (2), 296–302.

Di Stefano, L., Marchionnia, M., Mattoccia, S., 2004. A fast area-based stereo matching algorithm. Image Vision Comput. 22, 983–1005.

Faugeras, O., 1993. Three-Dimensional Computer Vision, A Geometric Viewpoint. The MIT Press.

Fernández, M.A., Fernández-Caballero, A., López, M.T., Mira, J., 2003. Length-speed ratio (LSR) as a characteristic for moving elements real-time classification. Real-Time Imaging 9, 49–59.

Fernández-Caballero, A., Fernández, M.A., Mira, J., Delgado, A.E., 2003a. Spatio-temporal shape building from image sequences using lateral interaction in accumulative computation. Pattern Recognition 36 (5), 1131–1142.

Fernández-Caballero, A., Mira, J., Fernández, M.A., Delgado, A.E., 2003b. On motion detection through a multi-layer neural network architecture. Neural Networks 16 (2), 205–222.

Fernández-Caballero, A., Mira, J., Delgado, A.E., Fernández, M.A., 2003c. Lateral interaction in accumulative computation: A model for motion detection. Neurocomputing 50, 341–364.

Fernández-Caballero, A., López, M.T., Fernández, M.A., Mira, J., Delgado, A.E., López-Valles, J.M., 2004. Accumulative computation method for motion features extraction in dynamic selective visual attention. In: Proc. 2nd International Workshop on Attention and Performance in Computational Vision, Lecture Notes in Computer Science 3368, 206–215.

Grimson, W.E.L., 1985. Computational experiments with a feature based stereo algorithm. IEEE Trans. Pattern Anal. Machine Intell. 7, 17–34.

Gutiérrez, S., Marroquín, J.L., 2004. Robust approach for disparity estimation in stereo vision. Image Vision Comput. 22 (3), 183–195.

Haralick, R., Shapiro, L., 1992. Computer and Robot Vision. Addison-Wesley.

Ho, A.Y.K., Pong, T.C., 1996. Cooperative fusion of stereo and motion. Pattern Recognition 29 (1), 121–130.

Koenderink, J., van Doorn, A., 1976. Geometry of binocular vision and a model for stereovision. Biol. Cybernetics 21, 29–35.

Liu, J., Skerjane, R., 1993. Stereo and motion correspondence in a sequence of stereo images. Signal Processing: Image Commun. 5 (4), 305–318.

López, M.T., Fernández-Caballero, A., Fernández, M.A., Mira, J., Delgado, A.E., 2006. Motion features to enhance scene segmentation in active visual attention. Pattern Recognition Letters 27 (5), 469–478.

Muhlmann, K., Maier, D., Hesser, J., Manner, R., 2002. Calculating dense disparity maps from color stereo images, an efficient implementation. Internat. J. Comput. Vision 47 (1–3), 79–88.

Scharstein, D., Szeliski, R., 2002. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. Internat. J. Comput. Vision 47 (1–3), 7–42.

Sumi, Y., Kawai, Y., Yoshimi, T., Tomita, F., 2002. 3D object recognition in cluttered environments by segment-based stereo vision. Internat. J. Comput. Vision 46 (1), 5–23.

Venkateswar, V., Chellappa, R., 1995. Hierarchical stereo and motion correspondence using feature groupings. Internat. J. Comput. Vision 15, 245–269.

Wildes, R., 1991. Direct recovery of three-dimensional scene geometry from binocular stereo disparity. IEEE Trans. Pattern Anal. Machine Intell. 13 (8), 761–774.

Wilson, R., Knutsson, H., 1989. A multiresolution stereovision algorithm based on Gabor representation. In: Proc. IEE Internat. Conf. on Image Processing and Appl., pp. 19–22.

Xu, G., 1995. Unification of stereo, motion and object recognition via epipolar geometry. In: Proc. 2nd Asian Conference on Computer Vision I, pp. 287–291.

Yang, W., Ngan, K., Lim, J., Sohn, K., 2005. Joint motion and disparity fields estimation for stereoscopic video sequences. Signal Processing: Image Commun. 20 (3), 265–276.

Zabih, R., Woodfill, J., 1994. Non parametric local transforms for computing visual correspondence. In: Proc. Third European Conference on Computer Vision, pp. 150–158.

Zhang, L., Curless, B., Seitz, S.M., 2003. Space–time stereo: Shape recovery for dynamic scenes. In: Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 367–374.